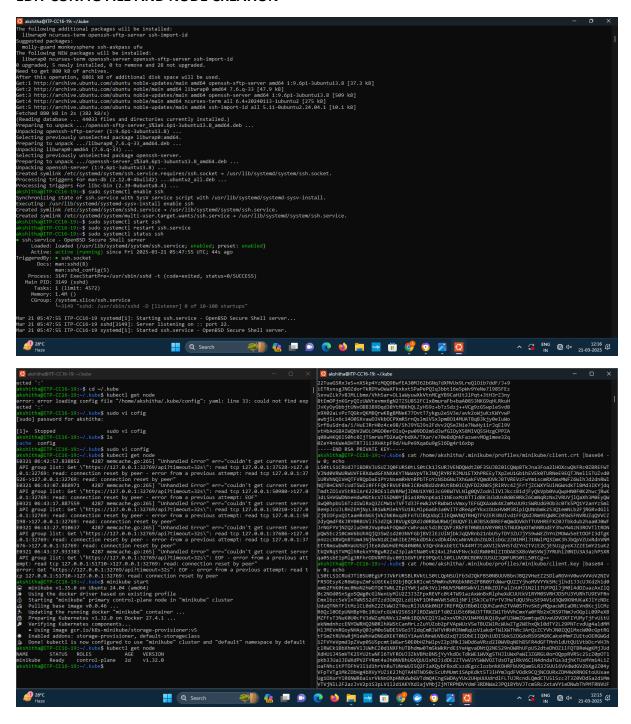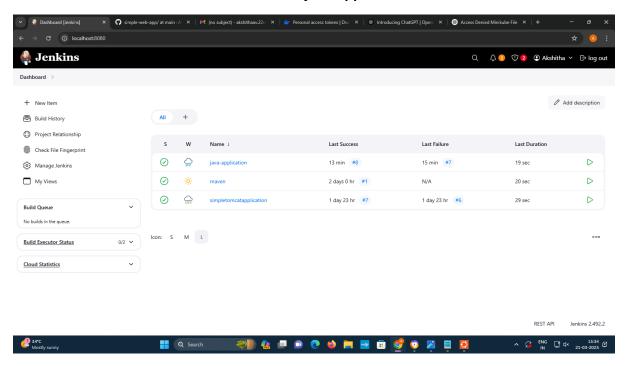# FINAL PROJECT

DAY-06

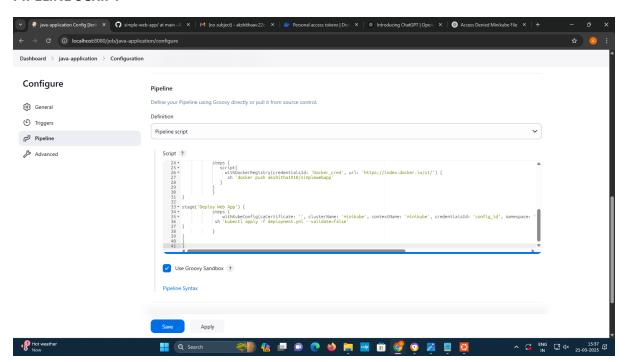# JAVA-APPLICATION DEPLOYMENT IN MINIKUBE

EDIT CONFIG FILE AND NODE CREATION

CREATION OF A PIPELINE PROJECT NAMED "java-application"

PIPELINE SCRIPT

PIPELINE SCRIPT CODE

```
pipeline {
    agent any

    stages {
        stage('scm') {
            steps {
            git branch: 'main', url: 'https://github.com/Akshi1910/simple-web-app.git'
            }
        }
        stage('build') {
            steps {
                sh "mvn clean"
                sh "mvn install"
}
}
stage('build to images') {
        steps {
          script{
            sh 'docker build -t akshitha1910/simplewebapp .'
          }
    }
}
stage('push to hub') {
        steps {
          script{
            withDockerRegistry(credentialsId: 'Docker_cred', url: 'https://index.docker.io/v1/') {
              sh 'docker push akshitha1910/simplewebapp'
          }
        }
        }
}

stage('Deploy Web App') {
        steps {
            withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName: 'minikube',
credentialsId: 'config_id', namespace: '', restrictKubeConfigAccess: false, serverUrl:
'https://192.168.39.226:8443') {
        sh 'kubectl apply -f deployment.yml --validate=false'
}
        }
}
```

```
}
}
```

DEPLOYMENT FILE

deployment.yml

```yaml
apiVersion: apps/v1

kind: Deployment

metadata:

  name: webnginx2

  labels:

    name: webnginx2

spec:

  replicas: 1

  selector:

    matchLabels:

      apptype: web-backend

  strategy:

    type: RollingUpdate

  template:

    metadata:

      labels:

        apptype: web-backend

    spec:

      containers:

      - name: webnginx2

        image: akshitha1910/simplewebapp:latest

        ports:

        - containerPort: 7070

---
```

```yaml
apiVersion: v1

kind: Service

metadata:

  name: my-service

  labels:

    app: my-service

    type: backend-app

spec:

  type: NodePort

  ports:

   - targetPort: 8080

     port: 7070

     nodePort: 30002

  selector:

    apptype: web-backend
```

deployment.yml file should be present in the project:

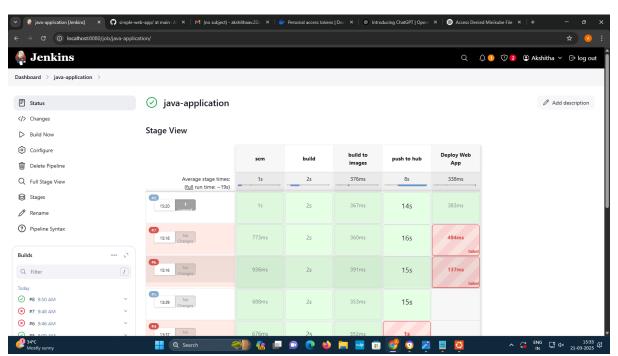gitlink: https://github.com/Akshi1910/simple-web-app
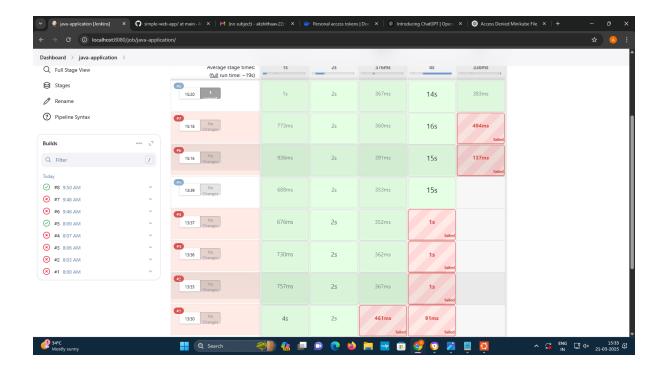
CREATION OF CREDENTIALS
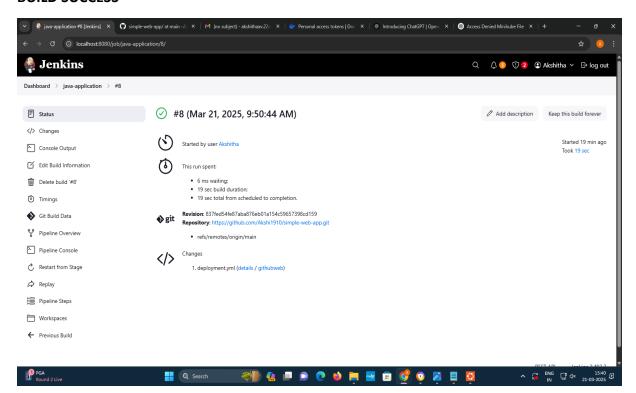
CONNECTING TO KUBERNETES

STAGE VIEW OF BUILD

BUILD SUCCESS

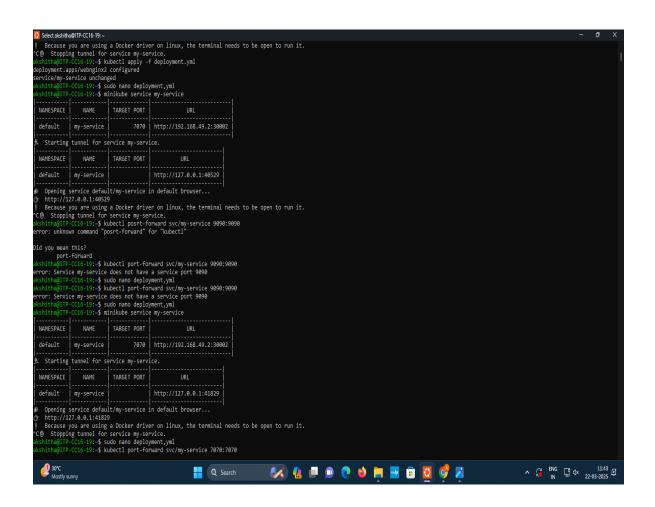**DEPLOYMENT FROM MINIKUBE(TERMINAL)**
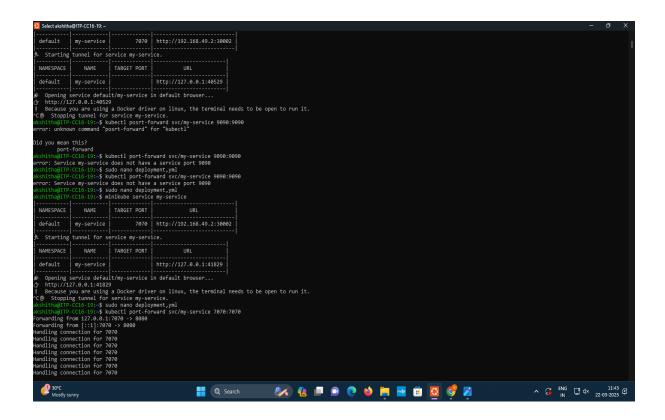
Commands:

minikube start

sudo nano deployment.yml

kubectl apply -f deployment.yml

kubectl get pod

minikube service my-service

kubectl port-forward svc/my-service 7070:7070

```
| default  | my-service |        7070 | http://192.168.49.2:30002 |
|----------|------------|-------------|---------------------------|
⬈  Starting tunnel for service my-service.
|-----------|------------|--------------|---------------------------|
| NAMESPACE |    NAME    | TARGET PORT  |           URL             |
|-----------|------------|--------------|---------------------------|
| default   | my-service |              | http://127.0.0.1:40529    |
|-----------|------------|--------------|---------------------------|
🎉  Opening service default/my-service in default browser...
👉  http://127.0.0.1:40529
❗  Because you are using a Docker driver on linux, the terminal needs to be open to run it.
^C🛑  Stopping tunnel for service my-service.
akshitha@ITP-CC16-19:~$ kubectl posrt-forward svc/my-service 9090:9090
error: unknown command "posrt-forward" for "kubectl"

Did you mean this?
        port-forward
akshitha@ITP-CC16-19:~$ kubectl port-forward svc/my-service 9090:9090
error: Service my-service does not have a service port 9090
akshitha@ITP-CC16-19:~$ sudo nano deployment,yml
akshitha@ITP-CC16-19:~$ kubectl port-forward svc/my-service 9090:9090
error: Service my-service does not have a service port 9090
akshitha@ITP-CC16-19:~$ sudo nano deployment,yml
akshitha@ITP-CC16-19:~$ minikube service my-service
|-----------|------------|--------------|---------------------------|
| NAMESPACE |    NAME    | TARGET PORT  |           URL             |
|-----------|------------|--------------|---------------------------|
| default   | my-service |        7070  | http://192.168.49.2:30002 |
|-----------|------------|--------------|---------------------------|
⬈  Starting tunnel for service my-service.
|-----------|------------|--------------|---------------------------|
| NAMESPACE |    NAME    | TARGET PORT  |           URL             |
|-----------|------------|--------------|---------------------------|
| default   | my-service |              | http://127.0.0.1:41829    |
|-----------|------------|--------------|---------------------------|
🎉  Opening service default/my-service in default browser...
👉  http://127.0.0.1:41829
❗  Because you are using a Docker driver on linux, the terminal needs to be open to run it.
^C🛑  Stopping tunnel for service my-service.
akshitha@ITP-CC16-19:~$ sudo nano deployment,yml
akshitha@ITP-CC16-19:~$ kubectl port-forward svc/my-service 7070:7070
Forwarding from 127.0.0.1:7070 -> 8080
Forwarding from [::1]:7070 -> 8080
Handling connection for 7070
Handling connection for 7070
Handling connection for 7070
Handling connection for 7070
Handling connection for 7070
Handling connection for 7070
Handling connection for 7070
```

On browser run https://127.0.0.1:7070/maven-web-app

Hello World!